REMARKS

Status of the Claims

Claims 1-58 are pending in the present application. No amendments to the claims have been made. The title has been amended to more clearly indicate the nature of the invention, and minor changes have been made to one of the paragraphs in the specification.

Telephone Interview with Examiner Ali

On March 10, 2005, applicants' attorney discussed the current Office Action with Examiner Ali during a telephone interview. The discussion focused on the Peterson reference, which describes a network environment including a plurality of servers, where each server appears to include a copy of the same version of a server-hosted application. Applicants' attorney noted that in the context of the present patent application and the claims included therein, the network environment includes a plurality of servers, and at least one server includes a different *version* of the server-hosted application. Examiner Ali noted that a portion of the Peterson reference specifically discusses updating information on Peterson's servers. Because Peterson clearly talks about updating different servers in Peterson's network, the Examiner concluded that updating Peterson's servers, each of which includes a copy of the same server-hosted application, is equivalent to installing multiple *versions* of a server-hosted application in a network environment.

Applicants' attorney then pointed out that updating data is different than installing a new version of a software application. Examiner Ali appeared to agree, but stated that the term "software application" was not included in the claims as filed. Applicants' attorney suggested that the term "server-hosted application" as used in the present application and claims thereof ought to convey to one of ordinary skill in the art that the server-hosted application is indeed a software application. No agreement on whether one of ordinary skill in the art would have implicitly have understood a server-hosted application to be a software application was reached.

Examiner Ali seemed to indicate that amending the claims to recite a "server-hosted *software* application," as opposed to "server-hosted application," would appear to distinguish over the cited art. The Examiner noted that if such an amendment were filed, an additional search would be performed for art related to server-hosted software applications, and if the additional search did not yield any relevant references, there was a reasonable likelihood that the case would be allowed.

Applicants' attorney would like to thank Examiner Ali for his time and willingness to discuss these issues during telephone interview.

Objection to Title

The Examiner has objected to the title because it is not descriptive of the invention. Applicants have amended the title, and therefore the objection to the title should be withdrawn.

Objection to the Specification

The Examiner has objected to the specification because it includes an embedded hyperlink. The hyperlink has been deleted as indicated in the above amendment to the specification. Accordingly, the objection to the specification should be withdrawn.

Objection to Claims 17, 34, and 57

The Examiner has objected to Claims 17, 34, and 57, indicating that such claims (each of which are directed to a machine readable medium) should be written as independent claims, rather than depending from a method claim. Applicants have successfully prosecuted to issue many applications in which dependent claims directed to memory media and drafted in the fashion of Claims 17, 34, and 57 were allowed, and applicants are not aware of any portion of the MPEP or the statutes that precludes dependent claims directed to machine instructions stored on such memory media that depend from a method claim. Accordingly, applicants believe that Claims 17, 34, and 57 can remain as dependent claims in conformance with acceptable practice. It is respectfully suggested that the Examiner discuss this matter with his Supervising Examiner, who may provide further insight. However, if the Examiner can cite a statutory basis for requiring that such claims be written in independent form, applicants will so amend Claims 17, 34, and 57. At this point, applicants respectfully request the Examiner either withdraw the objection to Claims 17, 34, and 57, or cite a statutory basis for the objection.

Problem Solved by the Present Invention

Before discussing the substantive rejection of the claims, it will be useful to briefly articulate the problem the present invention was intended to solve. It will be appreciated that software applications are frequently revised, particularly during a development phase. Such revisions add new capabilities, add new functionalities, and/or correct errors (i.e., bugs) discovered in earlier versions. In the past, server-hosted software applications have been revised by taking the server-hosted software application off-line (thereby preventing clients from accessing the application), updating the software application, migrating all data associated with the software application to the new version of the software application, and restarting the software application (enabling clients to access the updated application). This technique is described in the Background of the Invention section of applicants' specification.

The present invention provides an alternative approach for updating a server-hosted software application. Essentially, in the present invention, instead of taking a previous version of a software application off-line, a different version of the same software application is loaded onto one of the network servers. Initially, a small number of clients are directed to the different version of the server-hosted software application. If those clients do not report any problems with the new version of the software application, additional clients are migrated to the different version of the software application. This technique is thus less disruptive, because a version of the software application is always online and available for client use, and because initially, only the first few clients who are enabled to access the different version of the software application can be affected, if the new version of the software application fails to perform properly or causes undesired results.

Claims Rejected under 35 U.S.C. § 103

The Examiner has rejected Claims 1-58 as being obvious over Peterson (USPGPUB 2002/0103907) in view of Pezutti (USPGPUB 2004/0249927). The Examiner asserts that Peterson discloses most of the elements of the claimed invention, and that Pezutti discloses each remaining element that is not disclosed by Peterson. The Examiner asserts that it would been obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited art, because such a combination would have provided network access services for the benefit of network providers, service providers, and customers. The Examiner further notes that Pezutti suggests that installation of his invention improves service to customers, network providers and service providers, which provides motivation to modify Peterson in view of Pezutti to achieve applicants' claimed invention. Applicants respectfully disagree with this conclusion for the following reasons.

While the Examiner did note in the above-referenced telephone interview that replacing the term *server-hosted application* with the term *server-hosted software application* would appear to more clearly distinguish the present invention over the cited art, after much consideration, applicants have elected to forego such an amendment at this time, because given the context of the application, and the normal meaning of the term *server-hosted application*, one of ordinary skill at the time the application was filed should clearly have understood that a *server-hosted application* is indeed a software application. Accordingly, this change to the claim language does not seem to be justified. The term “updating” as used in Peterson is clearly directed to updating a set of data, as opposed to updating a version of a software application, as discussed in greater detail below. Thus, the cited art does not present a *prima facie* case of obviousness, because the cited art does not teach or suggest a network including two or more different ***versions*** of a ***server-hosted application*** that are operating concurrently.

The *Free Online Dictionary of Computing* (FOLDOC), a reference with which those of ordinary skill in the computing arts are likely to be acquainted, provides the following definition for the term “*application*:”

> A complete, self-contained program that performs a specific function directly for the user. This is in contrast to system software such as the operating system kernel, server processes and libraries which exists to support application programs. Editors for various kinds of documents, spreadsheets, and text formatters are common examples of applications. Network applications include clients such as those for FTP, electronic mail, telnet and WWW. The term is used fairly loosely, for instance, some might say that a client and server together form a distributed application; others might argue that editors and compilers were not applications but tools for building applications. One distinction between an application program and the operating system is that applications always run in "user mode" (or "non-privileged mode"), while operating systems and related utilities may run in "supervisor mode" (or "privileged mode"). The term may also be used to distinguish programs which communicate via a graphical user interface from those which are executed from the command line. (1994-11-28)

Significantly the *FOLDOC* entry for the term *application* appears to date from 1994, well before the filing of the present application. It appears that at least as early as 1994, one of ordinary skill in the computer arts would have recognized that the term *application* referred to a type of *software* (such as document editing and spreadsheet programs) distinguishable from *system software* (such as operating systems).

The *FOLDOC* provides the following definition for the term *software*:

> The instructions executed by a computer, as opposed to the physical device on which they run (the "hardware"). The term was coined by the eminent statistician, John Tukey. Programs stored on non-volatile storage built from

> integrated circuits (e.g. ROM or PROM) are usually called firmware. Software can be split into two main types - system software and application software or application programs. System software is any software required to support the production or execution of application programs but which is not specific to any particular application. Examples of system software would include the operating system, compilers, editors and sorting programs. Examples of application programs would include an accounts package or a CAD program. Other broad classes of application software include real-time software, business software, scientific and engineering software, embedded software, personal computer software and artificial intelligence software. Software includes both source code written by humans and executable machine code produced by assemblers or compilers. It does not usually include the data processed by programs unless this is in a format such as multimedia which depends on the use of computers for its presentation. This distinction becomes unclear in cases such as spread sheets which can contain both instructions (formulae and macros) and data. There are also various intermediate compiled or semi-compiled, forms of software such as library files and byte-code. Some claim that documentation (both paper and electronic) is also software. Others go further and define software to be programs plus documentation though this does not correspond with common usage. The noun "program" describes a single, complete and more-or-less self-contained list of instructions, often stored in a single file, whereas "code" and "software" are uncountable nouns describing some number of instructions which may constitute one or more programs or part thereof. Most programs, however, rely heavily on various kinds of operating system software for their execution. The nouns "code" and "software" both refer to the same thing but "code" tends to suggest an interest in the implementation details whereas "software" is more of a user's term. (2002-07-21)

Note the distinction between *system software* (software required to support the production or execution of application programs but which is not specific to any particular application, such as operating system, compilers, editors and sorting programs) and *application programs* (such as a CAD program). While the *FOLDOC* entry for *software* appears to several months later than the filing date of the present application, there is no basis for concluding that the definition of the term "*software*" underwent any significant change in these few short months, particularly given the maturity of the computing arts. Also, the definition is entirely consistent with the *FOLDOC* entry for *application* (provided above).

It is also noteworthy that the USPTO Classification relating to networked environments (Class 709) specifically defines the term application program as follows:

> A computer program designed to perform a certain type of work, such as ***an application*** to manipulate text, numbers, graphics, or a combination of these elements. ***An application*** differs from an operating system (which runs a computer), a utility (which performs maintenance or general-purpose chores), and a language (with which computer programs are created; *USPTO Patent Classification, Class Definitions, Class 709, Section III, Glossary*).

Note that the Patent Office itself appears to recognize that in the context of the computer arts, and computer networks in particular, an application is inherently software.

There does not appear to be any basis for concluding that the plain meaning of the term *server-hosted application* could have been interpreted by one of ordinary skill in the art at the time of the invention as being anything other than a software application (such as a CAD program, a word processing program, or an accounting program, as opposed to an operating system or a compiler). Indeed, the specification uses the term "*software*" no less than 77 times in describing the present invention. However, it is not necessary to use the term *software* with the term *application* in the claims, because one of ordinary skill in the art will clearly recognize that an application is indeed a type of software. In describing a working implementation of the present invention, the specification refers to FINANCE MANAGER as an exemplary server-hosted application. The specification specifically states that the invention is not intended to be limited only to the FINANCE MANAGER application, and that while it is likely that only two versions of software will be run simultaneously (an existing version and a new or upgrade version), it should be understood that, in accord with the present invention, more than two versions of a software application can be hosted at a network site at any one time (see page 10, second paragraph of applicants' specification). Note that the terms software and application are used in the same paragraph to refer to FINANCE MANAGER, which is simply an exemplary software/application.

Clearly, one of ordinary skill in the art at the time of the invention would have recognized that the term server-hosted application refers to software. If the Examiner disagrees with this conclusion, applicants respectfully request that the Examiner provide a definition for a server-hosted application that is not inherently a software application.

Turning now to the Peterson reference and what is disclosed by Peterson with respect to updating, it will be apparent that Peterson does not teach or suggest updating a version of a software application, but rather updating a field of data stored on a server. The portion of the Peterson reference including the term "update" is found in paragraph 66 of page 3, which is reproduced below.

> [0066] When vendor server 10 queries the network available devices on the server 10 network, or the devices report to the server (e.g., reporting can occur from device to vendor server to FAT, or through polling from FAT to server to device), a program residing on the devices issue a response to server 10. ***The information included in the response is then used to update the information stored on server 10 as to what resources (e.g. server, database, recordable medium, etc.) are available on the server 10 network (see FIG. 1a).*** When the central server 5 queries server 10, the program residing on the server 10 issues a response to the central server 5. The information included in the response is then used to update the information stored in the table. In an alternative embodiment, the servers 10-100 "log" onto the central server 5 and transmit information necessary to update the table (see FIG. 2). This embodiment will preferably be used when vendors register with the central server 5 for the first time. In this regard, each vendor registering with the central server 5 will report, for example, the corresponding IP address, storage and network capacity, and other information, which will then be stored in the table (see, for example, FIG. 4). The table is referred to as the File Allocation Table ("FAT"). Some of the information held in the table will be used to allocate data over the network to the server, depending on what is in the table. For example, the bandwidth capacity would be reported and stored in the table, as well as a calculation regarding what percentage of each servers network capacity is needed by the server for reasons other than the data storage service (see FIG. 4a). The table can also hold information identifying the location and ownership of data previously stored on each server. The table is then updated and revised as described above. The update takes place across various servers. Central server 5 is made up of several servers, dispersed over a network, such as the Internet, but connected to one another either over the network, or on their own network, for the purposes of mirroring the tables (the FAT tables) on each server providing the server 5 function. (Page 3, paragraph 66, emphasis added.)

While the art cited by the Examiner clearly involves a networked environment, it does not appear that the cited art addresses the problem addressed by the present invention, i.e., the problem involved with efficiently and smoothly migrating clients from one version of a server-hosted application to another version of the server-hosted application. Significantly, Peterson is directed to solving a different problem, i.e., allocating storage available on a plurality of networked servers so that clients can make efficient use of the storage resources over the network. In the context of Peterson, the term "updating" is clearly directed toward regularly determining the storage availability of the various storage resources available to the network, so that such resources can be allocated to clients. While it is true that Peterson teaches that each server with storage capacity is executing the same application (the Internet File Client application), Peterson does not teach or suggest that any of those multiple copies are ***different versions*** of the software application. Neither Peterson, nor any of the other cited art teaches or suggests updating a version of a server-hosted application, such that in a network, at least two different versions of the same server-hosted application are concurrently in use. The element of two different versions of the same server-hosted application to be simultaneously usable is a key advantage of applicants' claimed invention that is not taught or suggested by the cited art.

Pezutti appears to be related to a network architecture including customers, service retailers, service networks, and access networks and discusses networks that appear to be particularly well-suited to telecommunications. Significantly, it does not appear that Pezutti teaches or suggest any techniques useful to facilitate the migration of clients from one version of a server-hosted software application to a different version of the server-hosted software application in the manner recited by applicants' claims.

Because Peterson does not address or relate to the problem of how to update a server-hosted software application from a first version to a second version on the network, without adversely impacting clients, and because Pezutti similarly does not address the issue of migrating clients from one version of a server-hosted software application to another version of the server-hosted software application, there is no reason why one of ordinary skill in the art would be led to applicants' claimed invention in view of a combination of Peterson and Pezutti. These two references do not teach or suggest applicants' claimed invention. Accordingly, the rejection of Claims 1-58 as being obvious over Peterson in view of Pezutti should be withdrawn.

The preceding comments apply to each independent claim. It should be noted that applicants have not specifically discussed the patentability of each dependent claim over the cited art, and have chosen to forgo such an analysis in an effort to reduce the complexity and length of this response. That decision should not be construed as an admission that the dependent claims are not patentably distinguished over the cited art for additional reasons.

Claims 1, 18, 36 and 58 further distinguish over the cited art, because these claims recite a register that associates each client with a specific *version* of the server-hosted application, and changing that register each time a client is migrated from one *version* to another. The cited art does not teach or suggest simultaneously hosting multiple *versions* of the same server-hosted application, or any equivalent element. Claims 1, 18, 36 and 58 are thus distinguishable over the cited art for this additional reason.

In view of the amendments to the title and the specification, and the Remarks set forth above, it will be apparent that the claims in this application define a novel and non-obvious invention, and that the application is in condition for allowance and should be passed to issue without further delay. Should any further questions remain, the Examiner is invited to telephone applicants' attorney at the number listed below.

Respectfully submitted,

Michael C. King
Registration No. 44,832

MCK/RMA:lrg

I hereby certify that this correspondence is being deposited with the U.S. Postal Service in a sealed envelope as first class mail with postage thereon fully prepaid addressed to: Commissioner for Patents, Alexandria, VA 22313-1450, on April 5, 2005.

Date: April 5, 2005

Kathy